

**SISTEM ADMINISTRASI DOKUMEN BERBASIS WEB
PADA FORUM HUMAN CAPITAL INDONESIA**



**Disusun sebagai salah satu syarat menyelesaikan Program Studi Strata I pada
Jurusan Informatika Fakultas Komunikasi dan Informatika**

Oleh :

ARLIN WIDYA RAHAYU

L200170014

**PROGRAM STUDI INFORMATIKA
FAKULTAS KOMUNIKASI DAN INFORMATIKA
UNIVERSITAS MUHAMMADIYAH SURAKARTA**

2021

HALAMAN PERSETUJUAN

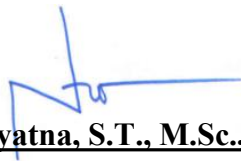
**SISTEM ADMINISTRASI DOKUMEN BERBASIS WEB
PADA FORUM HUMAN CAPITAL INDONESIA**

PUBLIKASI ILMIAH

oleh:

ARLIN WIDYA RAHAYU
L200170014

Telah diperiksa dan disetujui untuk diuji oleh:
Dosen Pembimbing



Nurgiyatna, S.T., M.Sc., Ph.D.
NIK. 881

HALAMAN PENGESAHAN

SISTEM ADMINISTRASI DOKUMEN BERBASIS WEB PADA FORUM HUMAN CAPITAL INDONESIA

OLEH

ARLIN WIDYA RAHAYU

L200170014

Telah dipertahankan di depan Dewan Penguji

Fakultas Komunikasi dan Informatika

Universitas Muhammadiyah Surakarta

Pada hari Senin, 5 April 2021

dan dinyatakan telah memenuhi syarat

Dewan Penguji:

1. Nurgiyatna, S.T., M.Sc., Ph.D.

(Ketua Dewan Penguji)

(.....)

2. Fatah Yasin Al Irsyadi, S.T., M.T.

(Anggota I Dewan Penguji)

(.....)

3. Dedi Gunawan, S.T., M.Sc., Ph.D.

(Anggota II Dewan Penguji)

(.....)

(.....)

Dekan

Fakultas Komunikasi dan Informatika



Nurgiyatna, S.T., M.Sc., Ph.D.
NIK. 881

PERNYATAAN

Dengan ini saya menyatakan bahwa dalam naskah publikasi ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu perguruan tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali secara tertulis diacu dalam naskah dan disebutkan dalam daftar pustaka.

Apabila kelak terbukti ada ketidakbenaran dalam pernyataan saya di atas, maka akan saya pertanggungjawabkan sepenuhnya.

Surakarta, 5 April 2021

Penulis

A handwritten signature in black ink, consisting of a large, stylized 'A' followed by a series of loops and a final vertical stroke.

ARLIN WIDYA RAHAYU

L200170014

SISTEM ADMINISTRASI DOKUMEN BERBASIS WEB PADA FORUM HUMAN CAPITAL INDONESIA

Abstrak

Forum Human Capital Indonesia (FHCI) merupakan komunitas para Pengelola Manajemen *Human Capital* BUMN untuk saling berinteraksi dan melakukan pembelajaran dalam rangka mengembangkan manajemen *Human Capital* di Indonesia. Dalam menjalankan kegiatan operasional, *FHCI* perlu melakukan pengelolaan dokumen untuk mendukung kelancaran administrasinya. Pengelolaan suatu dokumen di *FHCI* dibedakan sesuai dengan jenis dan kegunaannya, seperti dokumen surat masuk, surat keluar, *invoice*, kuitansi, surat permintaan pembayaran maupun surat perintah bayar. Selama ini pengarsipan dokumen hanya dilakukan melalui *Microsoft Excel*, hal ini menyulitkan para staf *FHCI* dalam mencari dokumen maupun menuliskan nomor urut dokumen. Maka dari itu, dibuatlah sistem administrasi dokumen yang bertujuan untuk memudahkan proses pengelolaan dan menghindari kesalahan penulisan nomor urut dokumen yang dikeluarkan. Dalam pengembangannya, sistem administrasi dokumen ini menerapkan metode *Software Development Life Cycle (SDLC)* yaitu menggunakan model pendekatan *Waterfall*. Sistem ini menggunakan bahasa pemrograman *PHP* dengan *Framework CodeIgniter* dan *database MySQL*. Hasil dari penelitian ini berupa *website* yang menampilkan data surat masuk, surat keluar, *invoice*, kuitansi, surat permintaan pembayaran dan surat perintah bayar serta dapat mencetak *template* surat keluar, *invoice*, kuitansi, surat permintaan pembayaran dan surat perintah bayar dengan penomoran yang otomatis. Untuk pengujiannya, sistem ini menggunakan *black box testing* dan *User Acceptance Test (UAT)* dengan hasil pengujian mencapai angka 91%.

Kata kunci : Administrasi Dokumen, *CodeIgniter*, Forum Human Capital Indonesia, *Waterfall*

Abstract

Forum Human Capital Indonesia (FHCI) is a community of Human Capital Management Managers to interact and learn to develop Human Capital management in Indonesia. In running its operational activities, FHCI need to manage a lot of documents to support its administration. The document management in FHCI is classified by its types and uses, such as received letter, sent letter, invoice, receipt, payment request letter, or a warrant of payment. FHCI has used Microsoft excel as its document administration system, but it is difficult for the staff to find and number the document. Therefore, this document administration system is created in order to expedite the management process and avoid the mistakes in its numbering system. This document administration system is developed by implementing the Software Development Life Cycle (SDLC) method with the Waterfall model approach. This system uses PHP programming language with Framework CodeIgniter and database MySQL. The result of this research is in the form of a website that shows the data of received letters, sent letters, invoice, receipt, payment request letter, and warrant of payment equipped with the automatic template and numbering system. Black Box Testing and User Acceptance Test (UAT) were used as testers with 91% results.

Keywords: Document Administration, CodeIgniter, Forum Human Capital Indonesia, Waterfall

1. PENDAHULUAN

Forum Human Capital Indonesia (FHCI) merupakan komunitas para Pengelola Manajemen *Human Capital* BUMN untuk saling berinteraksi, melakukan pembelajaran maupun sinergi kepada para anggotanya dalam rangka mengembangkan manajemen *Human Capital* di Indonesia. Dalam menjalankan kegiatan operasional, *FHCI* perlu melakukan pengelolaan dokumen untuk mendukung kelancaran administrasinya. Ketertiban dalam pengelolaan dokumen sangat berpengaruh dalam ketertiban administrasi suatu divisi maupun instansi (Setyawati, 2018).

Pengelolaan suatu dokumen di *FHCI* dibedakan sesuai dengan jenis dan kegunaannya, seperti dokumen surat masuk, surat keluar, *invoice*, kuitansi, surat permintaan pembayaran maupun surat perintah bayar. Selama ini, proses pengelolaan dokumen hanya dilakukan melalui *Microsoft Excel*, hal ini menjadi kendala tersendiri dalam proses pengarsipan dan pencarian dokumen. Selain itu, ada juga dokumen yang hilang, tidak diarsipkan ataupun penulisan nomor dokumen/surat yang *double* karena seiring bertambahnya dokumen/surat yang akan dikirimkan.

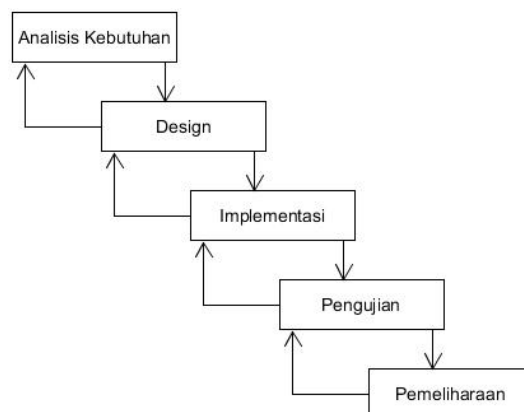
Dengan adanya permasalahan tersebut, maka dirancanglah sistem administrasi dokumen yang berbasis *web* untuk membantu aktivitas pengarsipan dokumen/surat. Sistem administrasi dokumen ini bisa menyimpan dokumen/surat dalam bentuk *digital* untuk menghemat ruang penyimpanan, menjamin keamanan dokumen/surat dan mengefisienkan kinerja pegawai *FHCI*. Selain itu, sistem ini bisa menghindari penulisan nomor dokumen yang *double/ganda*, serta mempermudah proses pencarian dokumen.

Hypertext Preprocessor (PHP) yang dapat mendefinisikan, membuat, menyimpan dan memanipulasi *database* di sisi *server* digunakan untuk mengimplementasikan sistem administrasi dokumen ini (Chandra, Asadi, & Raju, 2016). *PHP* menjadi bahasa skrip terkenal yang sering dikaitkan dengan pengembangan *web*, sehingga ada banyak *framework* yang muncul selama beberapa dekade terakhir (Solanki, Shah, & Shah, 2017). *Framework* yang digunakan untuk membangun sistem administrasi dokumen ini adalah *Framework CodeIgniter*. Dalam pengembangan aplikasi web standar *framework CodeIgniter* menerapkan konsep *Model, View and Controller (MVC)* dengan metodologi akses basis data (Vidal-Silva, Jiménez, Madariaga, & Urzúa, 2020). Pemisahan model, tampilan serta kontrol arsitektur aplikasi web inilah yang menggunakan konsep *MVC*. Sedangkan untuk

pengelolaan *database* digunakanlah *MySQL* yang merupakan *database* relasional untuk menyimpan data dalam struktur tabel dan dikelompokkan berdasarkan tabel tersebut (Andersson & Berggren, 2017).

2. METODE

Dalam pengembangannya, sistem administrasi dokumen menerapkan salah satu model dari metodologi siklus berulang untuk menjelaskan mengenai tahapan dalam pengembangan, perancangan dan pemeliharaan *software* yang biasa dikenal dengan metode *Software Development Life Cycle (SDLC)* yang menggunakan model pendekatan *Waterfall* agar sasaran, tujuan serta syarat-syarat fungsional dari suatu *software* dapat dicapai (Barjitya, Sharma, & Rani, 2017). *Waterfall* atau model air terjun adalah model linier dan berurutan dimana setiap tahapan harus diselesaikan secara lengkap sebelum melanjutkan ke tahap selanjutnya (Rahayu, 2020). Model *waterfall* terdiri dari beberapa tahap pengembangan perangkat lunak yaitu analisis kebutuhan, *design*, implementasi, pengujian dan pemeliharaan. Langkah-langkah model pendekatan *waterfall* ditunjukkan pada Gambar 1.



Gambar 1. Langkah-langkah model pendekatan *waterfall* (Hardyanto, Purwinarko, Sujito, & Alighiri, 2017)

2.1 Analisis Kebutuhan

Tahap analisis kebutuhan ini dimulai dengan mengumpulkan data-data kebutuhan dari sistem yang akan dikembangkan. Dalam mengumpulkan data terkait kebutuhan fungsional serta non-fungsional sistem administrasi dokumen, maka dilakukan wawancara/diskusi dengan para staf kesekretariatan dan bendahara FHCI agar menghasilkan sistem yang sesuai dengan kebutuhan.

2.1.1 Analisis Kebutuhan Fungsional

Pada sistem administrasi dokumen terdapat beberapa kebutuhan

fungsional, yaitu:

- 1) Sistem ini dapat melakukan *login multilevel* serta *logout*.
- 2) Dalam pengelolaan data, sistem ini terbagi menjadi dua hak akses yang berupa *root* dan *admin* dengan wewenang masing-masing.
- 3) Sistem dapat manajemen data divisi, data kode surat, data surat bentuk khusus, data surat masuk, data surat keluar, data vendor, data *invoice*, data kuitansi, data surat permintaan pembayaran dan data surat perintah bayar (menambah, menghapus, mengubah dan menampilkan data).
- 4) Sistem dapat mengunggah dan mengunduh *file* surat bentuk khusus, surat masuk, surat keluar, *invoice*, kuitansi, surat permintaan pembayaran dan surat perintah bayar.
- 5) Sistem dapat mencetak *template* surat keluar, *invoice*, kuitansi, surat permintaan pembayaran dan surat perintah bayar dengan penomoran dokumen yang otomatis.

2.1.2 Analisis Kebutuhan Non-fungsional

Pada sistem administrasi dokumen terdapat beberapa kebutuhan non-fungsional, yaitu:

- 1) Kebutuhan perangkat keras (*hardware*)
Seperangkat komputer yang digunakan untuk memasukkan data baru ke sistem administrasi dokumen *FHCI*.
- 2) Kebutuhan perangkat lunak (*software*)
Software yang diperlukan dalam pengembangan sistem administrasi dokumen ini diantaranya adalah *Windows Operating System* (7,8,10), *XAMPP*, *phpMyAdmin*, *Navicat Premium 12*, *text editor Visual Studio Code*, menggunakan *web browser Google Chrome* dan *Framework CodeIgniter (CI)*.

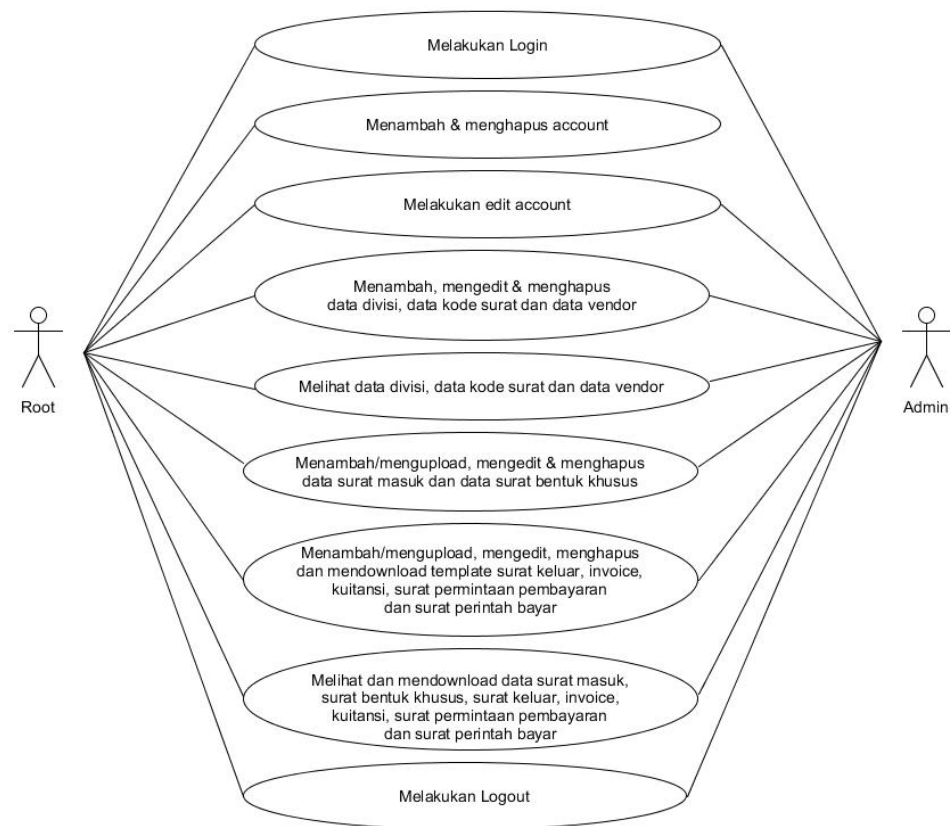
2.2 Design

Pada tahap ini perancangan sistem dibuat sesuai dengan spesifikasi kebutuhan yang diperoleh pada tahap analisis kebutuhan. Perancangan dimulai dengan penggambaran interaksi *actor* dengan sistem pada *use case* dalam permodelan *Unified Modelling Language (UML)* (Maylawati, Darmalaksana, & Ramdhani, 2018). Selain

itu, dalam pengembangan arsitektur sistem juga dilakukan perancangan basis data pada sistem administrasi dokumen yang akan dikembangkan.

2.2.1 Use Case Diagram

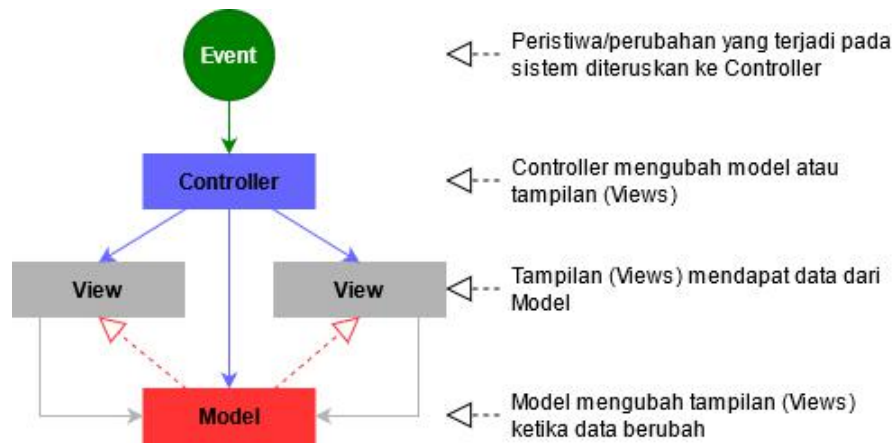
Object Oriented Diagrams yang menampilkan secara detail mengenai interaksi dalam sistem secara internal dan bagaimana lingkungan eksternal diatur digambarkan dalam suatu *Use Case Diagram* (Aleryani, 2016). Terdapat dua *actor* dengan masing-masing interaksinya pada *use case diagram* sistem administrasi dokumen *FHCI* ini, yang ditunjukkan pada Gambar 2.



Gambar 2. *Use Case Diagram* Sistem Administrasi Dokumen *FHCI*

2.2.2 Perancangan Database

Dalam perancangan basisdata biasanya melibatkan analisis domain masalah dan menghasilkan diagram *database* menggunakan model *Entity Relationship Diagram* yang kemudian diubah menjadi skema relasional sebelum *database* fisik dibuat dan diimplementasikan (Yang & Cao, 2016). Perancangan *database* pada sistem administrasi dokumen *FHCI* dapat dilihat pada Gambar 3.



Gambar 4. Arsitektur konsep *Model, View and Controller (MVC)*

2.4 Pengujian

Tahap *testing* atau pengujian perangkat lunak bertujuan untuk menguji fungsionalitas dan mencari kesalahan sehingga bisa meminimalisir *bug* dalam perangkat lunak. Hal ini dilakukan agar kesalahan pada sistem administrasi dokumen bisa dideteksi dan dikoreksi secepatnya.

Black box testing yang mengacu pada validasi spesifikasi fungsionalitas *software* akan digunakan untuk menguji sistem administrasi dokumen *FHCI* ini (Hidayat & Muttaqin, 2020). Pengujian dengan *black box testing* dapat mendeteksi kesalahan terkait struktur data, terminasi, inisialisasi serta kesalahan akses ke *database* (Finanda & Fatmawati, 2019). Selain *black box testing*, dilakukan juga *User Acceptance Test (UAT)* yang ditujukan kepada *user* untuk mengetahui tingkat kelayakan *software*.

2.5 Pemeliharaan

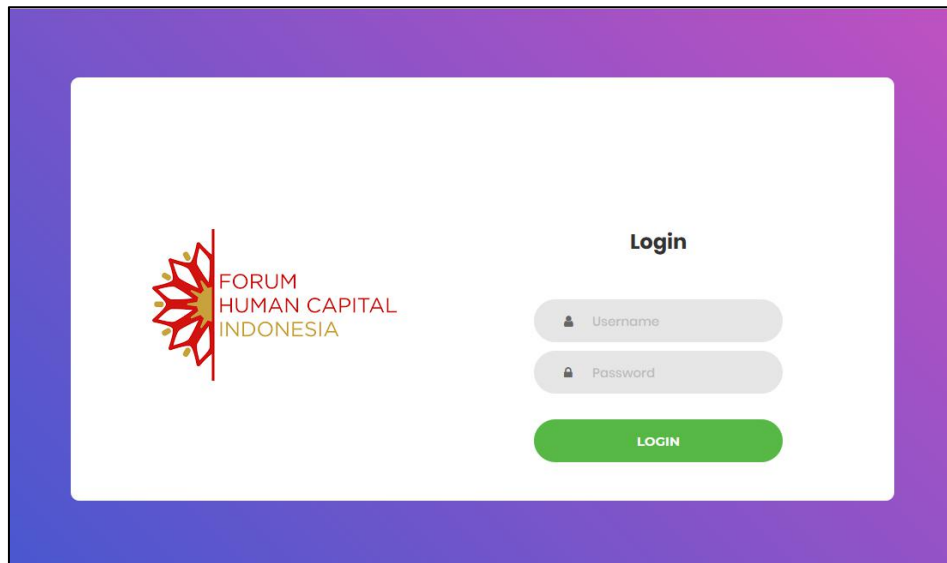
Pada tahap ini, sistem administrasi dokumen akan diinstal dan digunakan. Agar sistem dapat dimanfaatkan secara maksimal akan dilakukan perbaikan jika ada *error* yang tidak terdeteksi pada tahap pengujian serta menambahkan data sesuai kebutuhan *user* (Pramitasari & Nurgiyatna, 2019).

3. HASIL DAN PEMBAHASAN

3.1 Hasil

A. Halaman *Login Multilevel*

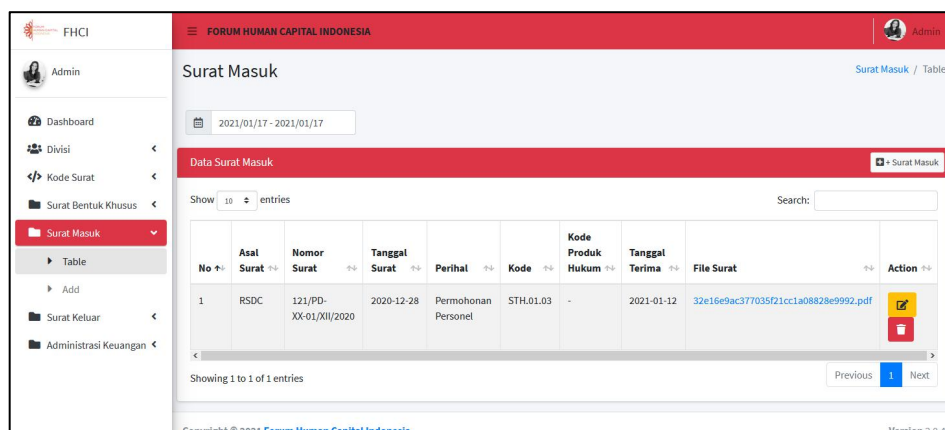
Halaman *login multilevel* adalah tampilan awal yang memuat *form* untuk memasukkan *username* serta *password* yang kemudian dilakukan autentikasi, agar pengguna dapat masuk ke dalam sistem administrasi dokumen sesuai dengan hak aksesnya. Halaman *login multilevel* ditunjukkan pada Gambar 5.



Gambar 5. Halaman *Login Multilevel*

B. Halaman Data Surat Masuk

Halaman ini menampilkan tabel data surat masuk yang memungkinkan bagi *user* untuk menambah data, mengubah data, menghapus data maupun mengunggah *file* surat masuk. Halaman data surat masuk dapat dilihat pada Gambar 6.



Gambar 6. Halaman Data Surat Masuk

C. Halaman Tambah Data Surat Keluar

Halaman ini digunakan untuk menambahkan data surat keluar yang nantinya *draft* surat tersebut dapat diunduh pada fitur *download* yang terdapat pada halaman data surat keluar. Halaman tambah data surat keluar dapat dilihat pada Gambar 7.

The screenshot shows the 'Surat Keluar' form in the FHCI system. The form is titled '+ Surat Keluar' and includes a 'Form Input' section. The fields are as follows:

- Jenis Surat Keluar: Text input field.
- Tanggal Terbit: Date input field (mm / dd / yyyy).
- Tanggal Agenda: Date input field (mm / dd / yyyy).
- Perihal: Text input field.
- Tujuan: Text input field.
- Jabatan: Text input field.
- Perusahaan: Text input field.
- Divisi: Dropdown menu (--Divisi--).
- Kode Masalah: Dropdown menu (--Kode Masalah--).
- Kode Surat Bentuk Khusus: Dropdown menu.
- Kode Produk Hukum: Dropdown menu.

At the bottom of the form are three buttons: Cancel (yellow), Reset (blue), and Submit (green). The footer shows 'Copyright © 2021 Forum Human Capital Indonesia' and 'Version 3.0.4'.

Gambar 7. Halaman tambah data surat keluar

D. Halaman Tambah Data *Invoice*

Halaman tambah data *invoice* digunakan untuk membuat *draft invoice* dengan cara memasukkan data pada *form* tambah data *invoice*. Setelah itu, *draft invoice* tersebut dapat diunduh pada fitur *download* yang terdapat pada halaman data *invoice*. Halaman tambah data *invoice* dapat dilihat pada Gambar 8.

The screenshot shows the 'Invoice' form in the FHCI system. The form is titled '+ Invoice' and includes a 'Form Input' section. The fields are as follows:

- Vendor: Dropdown menu (--Vendor--).
- Divisi: Dropdown menu (--Divisi--).
- Kode Masalah: Dropdown menu (--Kode Masalah--).
- Tanggal: Date input field (mm / dd / yyyy).
- Perihal: Text input field.

Below the input fields is a table with the following structure:

Tabel	Description	Quantity	Price
	Description	Quantity	Price
	Description	Quantity	Price
	Description	Quantity	Price
	Description	Quantity	Price
	Description	Quantity	Price

At the bottom of the form are three buttons: Cancel (yellow), Reset (blue), and Submit (green). The footer shows 'Copyright © 2021 Forum Human Capital Indonesia' and 'Version 3.0.4'.

Gambar 8. Halaman Tambah Data *Invoice*

E. Halaman Tambah Data Kuitansi

Halaman ini digunakan untuk membuat *draft* kuitansi dengan cara memasukkan data pada *form* tambah data kuitansi. Setelah itu, *draft* kuitansi tersebut dapat diunduh pada fitur *download* yang terdapat pada halaman data kuitansi. Halaman tambah data kuitansi dapat dilihat pada Gambar 9.

The screenshot displays a web application interface for adding receipt data. The main content area is titled '+ Kuitansi' and contains a 'Form Input' section. The form includes the following fields: 'Vendor' (dropdown menu), 'Divisi' (dropdown menu), 'Kode Masalah' (dropdown menu), 'Tanggal' (date input field with format mm/dd/yyyy), 'Perihal' (text input field), 'Besaran' (text input field), and 'Keterangan' (text input field). Below the form are three buttons: 'Cancel' (yellow), 'Reset' (blue), and 'Submit' (green). The left sidebar shows a navigation menu with 'Administrasi Keuangan' highlighted, and sub-items include 'Vendor', 'Invoice', 'Kuitansi', 'Surat Permintaan Pembayaran', and 'Surat Perintah Bayar'. The top header shows the logo of FHCI and the text 'FORUM HUMAN CAPITAL INDONESIA'. The bottom of the page shows the copyright notice 'Copyright © 2021 Forum Human Capital Indonesia' and the version 'Version 3.0.4'.

Gambar 9. Halaman Tambah Data Kuitansi

F. Halaman Tambah Data Surat Permintaan Pembayaran

Halaman ini berfungsi untuk membuat *draft* surat permintaan pembayaran (spp) dengan cara memasukkan data pada *form*, kemudian *draft* spp tersebut dapat diunduh pada fitur *download* yang terdapat pada halaman data surat permintaan pembayaran. Halaman tambah data surat permintaan pembayaran dapat dilihat pada Gambar 10.

The screenshot shows the 'Surat Permintaan Pembayaran' (Request for Payment) form in the FHCI system. The form is titled '+ Surat Permintaan Pembayaran' and includes a sidebar with navigation options like Dashboard, Divisi, Kode Surat, Surat Bentuk Khusus, Surat Masuk, Surat Keluar, Administrasi Keuangan, Vendor, Invoice, Kuitansi, Surat Permintaan Pemb., and Surat Perintah Bayar. The form fields are as follows:

- Vendor: --Vendor--
- Divisi: --Divisi--
- Kode Masalah: --Kode Masalah--
- Tanggal: mm / dd / yyyy
- Divisi Vendor: Divisi Vendor
- Besaran: Besaran
- Bank: Bank
- Rekening: Rekening
- Atas Nama: Atas Nama
- Beban Anggaran Tahun: Beban Anggaran Tahun
- Nomor Referensi: Diisi dengan Nomor Justifikasi / Nomor Invoice
- Uraian Transaksi: Uraian Transaksi

At the bottom of the form are three buttons: Cancel (yellow), Reset (blue), and Submit (green). The footer shows 'Copyright © 2021 Forum Human Capital Indonesia.' and 'Version 3.0.4'.

Gambar 10. Halaman Tambah Data Surat Permintaan Pembayaran

G. Halaman Tambah Data Surat Perintah Bayar

Pada halaman ini pengguna dapat membuat *draft* surat perintah bayar (spb) dengan cara memasukkan data pada *form*, kemudian *draft* spb tersebut dapat diunduh pada fitur *download* yang terdapat pada halaman data surat perintah bayar. Halaman tambah data surat perintah bayar dapat dilihat pada Gambar 11.

The screenshot shows the 'Surat Perintah Bayar' (Payment Order) form in the FHCI system. The form is titled '+ Surat Perintah Bayar' and includes a sidebar with navigation options like Dashboard, Divisi, Kode Surat, Surat Bentuk Khusus, Surat Masuk, Surat Keluar, Administrasi Keuangan, Vendor, Invoice, Kuitansi, Surat Permintaan Pemb., and Surat Perintah Bayar. The form fields are as follows:

- No SPP: --No Surat Permintaan Pembayaran--
- Divisi: --Divisi--
- Kode Masalah: --Kode Masalah--
- Tanggal: mm / dd / yyyy
- Rekening Bayar: Rekening Bayar
- Jenis Transaksi: Jenis Transaksi
- Bank: Bank
- Rekening: Rekening
- Kepada: Kepada
- Jumlah: Jumlah
- Keterangan: Keterangan
- Catatan: Catatan

At the bottom of the form are three buttons: Cancel (yellow), Reset (blue), and Submit (green). The footer shows 'Copyright © 2021 Forum Human Capital Indonesia.' and 'Version 3.0.4'.

Gambar 11. Halaman Tambah Data Surat Perintah Bayar

3.2 Pengujian dan Pembahasan

A. Pengujian *Blackbox*

Pengujian *blackbox* bertujuan untuk memastikan bahwa setiap fitur dan fungsionalitas sistem berjalan sesuai dengan yang diharapkan. Hasil pengujian *blackbox* dapat dilihat pada Tabel 1.

Tabel 1. Hasil Pengujian *Blackbox*

No	Pengujian	Test Case	Hasil yang diharapkan	Hasil pengujian
1	<i>Login Multilevel</i>	Pengguna melakukan <i>input Username</i> dan <i>Password</i> dengan benar lalu menekan tombol " <i>Login</i> "	Masuk ke halaman <i>dashboard</i> sistem administrasi dokumen sesuai hak aksesnya	Sesuai
		Pengguna salah melakukan <i>input Username</i> dan <i>Password</i> lalu menekan tombol " <i>Login</i> "	Tetap berada di halaman <i>Login</i>	Sesuai
2	<i>Add Account</i>	Menambah data baru dengan cara mengisi <i>form input</i> , kemudian klik tombol " <i>Submit</i> "	Dapat menambah <i>account</i> baru dan menampilkan dalam tabel	Sesuai
3	<i>Add Divisi</i>	Menambah data baru dengan cara mengisi <i>form input</i> , kemudian klik tombol " <i>Submit</i> "	Dapat menambah data divisi baru dan menampilkan dalam tabel	Sesuai
4	<i>Add Kode Surat (Kode Surat Bentuk Khusus, Kode Produk Hukum, Kode Masalah)</i>	Menambah data baru dengan cara mengisi <i>form input</i> , kemudian klik tombol " <i>Submit</i> "	Dapat menambah data kode surat bentuk khusus/kode produk hukum/kode masalah dan menampilkan dalam tabel	Sesuai
5	<i>Add Surat Bentuk Khusus</i>	Menambah data baru dengan cara mengisi <i>form input</i> , kemudian klik tombol " <i>Submit</i> "	Dapat menambah data baru dan menampilkan dalam tabel	Sesuai
6	<i>Add Surat Masuk</i>	Menambah data baru dengan cara mengisi <i>form input</i> , kemudian klik tombol " <i>Submit</i> "	Dapat menambah data baru dan menampilkan dalam tabel	Sesuai
7	<i>Add Surat Keluar</i>	Menambah data baru dengan cara mengisi <i>form input</i> , kemudian klik tombol " <i>Submit</i> "	Dapat menambah data baru dan menampilkan dalam tabel	Sesuai
8	<i>Add Vendor</i>	Menambah data baru dengan cara mengisi <i>form input</i> , kemudian klik tombol " <i>Submit</i> "	Dapat menambah data baru dan menampilkan dalam tabel	Sesuai
9	<i>Add Invoice</i>	Menambah data baru dengan cara mengisi <i>form input</i> , kemudian klik tombol " <i>Submit</i> "	Dapat menambah data baru dan menampilkan dalam tabel	Sesuai
10	<i>Add Kuitansi</i>	Menambah data baru dengan cara mengisi <i>form input</i> , kemudian klik tombol " <i>Submit</i> "	Dapat menambah data baru dan menampilkan dalam tabel	Sesuai
11	<i>Add Surat Permintaan Pembayaran</i>	Menambah data baru dengan cara mengisi <i>form input</i> , kemudian klik tombol " <i>Submit</i> "	Dapat menambah data baru dan menampilkan dalam tabel	Sesuai
12	<i>Add Surat Perintah Bayar</i>	Menambah data baru dengan cara mengisi <i>form input</i> , kemudian klik tombol " <i>Submit</i> "	Dapat menambah data baru dan menampilkan dalam tabel	Sesuai
13	<i>Logout</i>	Menekan tombol " <i>Logout</i> " untuk keluar dari sistem	Keluar dari sistem dan kembali ke halaman <i>Login Multilevel</i>	Sesuai

B. Pengujian *User Acceptance Test (UAT)*

User Acceptance Test (UAT) adalah pengujian yang dilakukan oleh *end user* untuk memastikan bahwa sistem tersebut memenuhi proses bisnis sesuai dengan kebutuhan *end user* (Padmini, Perera, & Bandara, 2016). Pada pengujian *UAT* ini, dilakukan dengan metode *survey* dengan cara menyebar kuesioner kepada 5 orang staf kesekretariatan FHCI yang diberikan 10 pertanyaan dimana setiap jawaban memiliki bobot penilaian berbeda seperti yang dapat dilihat pada Tabel 2, nantinya jawaban responden akan dihitung menggunakan Rumus 1 (Audina & Supriyono, 2020). Hasil perhitungan nilai pengujian *UAT* dari jawaban responden dapat dilihat pada Tabel 3.

Tabel 2. Bobot penilaian jawaban responden

Opsi jawaban	Kode	Bobot	Persentase
Sangat Setuju	SS	5	81% - 100%
Setuju	S	4	61% - 80%
Netral	N	3	41% - 60%
Tidak Setuju	TS	2	21% - 40%
Sangat Tidak Setuju	STS	1	0% - 20%

Rumus 1. Rumus perhitungan hasil persentase pengujian *UAT*

$$\text{Hasil Persentase} = \left(\frac{\sum \text{Total Score}}{\sum \text{Jumlah Responden} \times 5} \right) \times 100\% \quad \dots\dots(1)$$

Tabel 3. Hasil perhitungan nilai pengujian

No	Pertanyaan	Nilai					Total Score	Hasil Persentase
		SSx5	Sx4	Nx3	TSx2	STSx1		
1	Tampilan sistem mudah dipahami	20	4				24	96%
2	Sistem ini menampilkan informasi yang mudah dimengerti	10	12				22	88%
3	Sistem mudah digunakan	25					25	100%
4	Fitur-fitur pada sistem ini sudah sesuai dengan kebutuhan	15	8				23	92%
5	Tidak terjadi <i>bug/error</i> /kesalahan pada sistem	10	8	3			21	84%
6	Sistem dapat menghasilkan <i>draft/template</i> dokumen yang fleksibel dan sesuai kebutuhan	15	8				23	92%
7	Sistem ini membantu dalam mempermudah kinerja pengguna	10	12				22	88%
8	Sistem ini dapat digunakan dan dikembangkan dalam jangka waktu dekat	20	4				24	96%
9	Keamanan data yang diinput terjamin dari orang yang tidak berhak mengakses	10	8	3			21	84%
10	Anda puas dengan penggunaan sistem tersebut	15	8				23	92%
								91%

Berdasarkan pengujian *User Acceptance Test* yang diperoleh dari jawaban responden menunjukkan bahwa hasil persentase pengujian sistem administrasi dokumen ini memperoleh nilai persentase 91% yang dapat disimpulkan bahwa sistem ini sangat disetujui dan dapat digunakan oleh *Forum Human Capital Indonesia*.

4. PENUTUP

Sistem administrasi dokumen berbasis *web* ini dapat mempermudah kinerja staf *FHCI* dalam mencari dokumen, mengelompokkan dokumen berdasarkan jenis dan kegunaannya serta meminimalisir kesalahan dalam penulisan nomor surat/dokumen. Berdasarkan hasil pengujian *User Acceptance Test (UAT)* yang mencapai nilai persentase 91%, sehingga dapat disimpulkan bahwa sistem ini dapat digunakan dan sesuai dengan kebutuhan *Forum Human Capital Indonesia*. Untuk kedepannya, sistem ini bisa dikembangkan dengan menambahkan fitur notifikasi apabila ada data yang belum terisi dengan lengkap.

DAFTAR PUSTAKA

- Andersson, E., & Berggren, Z. (2017). A Comparison Between MongoDB and MySQL Document Store Considering Performance.
- Aleryani, A. Y. (2016). Comparative study between data flow diagram and use case diagram. *International Journal of Scientific and Research Publications*, 6(3), 124-126.
- Audina, W., & Supriyono, H. (2020). Rancang Bangun Sistem Piket Kepegawaian Berbasis Web di PT Perusahaan Gas Negara Tbk (Doctoral dissertation, Universitas Muhammadiyah Surakarta).
- Barjtya, S., Sharma, A., & Rani, U. (2017). A detailed study of Software Development Life Cycle (SDLC) models. *International Journal Of Engineering And Computer Science*, 6(7).
- Chandra, D. S., Asadi, S. S., & Raju, M. V. S. (2016). DESIGN OF WEB BASED DECISION SUPPORT SYSTEM-MODEL STUDY OF VIJAYAWADA, AP. *Technology*, 7(6), 514-522.
- Destiningrum, M., & Adrian, Q. J. (2017). Sistem Informasi Penjadwalan Dokter Berbasis Web Dengan Menggunakan Framework Codeigniter (Studi Kasus: Rumah Sakit Yukum Medical Centre). *Jurnal Teknoinfo*, 11(2), 30-37.

- Finanda, F., & Fatmawati, A. (2019). Sistem Informasi Pengelolaan Pariwisata" SIPETA". *Jurnal INSYPRO (Information System and Processing)*, 4(2).
- Hardyanto, W., Purwinarko, A., Sujito, F., & Alighiri, D. (2017, April). Applying an MVC framework for the system development life cycle with waterfall model extended. In *Journal of Physics Conference Series* (Vol. 824, No. 1, p. 012007).
- Hidayat, T., & Muttaqin, M. (2020). Pengujian sistem informasi pendaftaran dan pembayaran wisuda online menggunakan black box testing dengan metode equivalence partitioning dan boundary value analysis. *Jutis (Jurnal Teknik Informatika)*, 6(1), 25-29.
- Maylawati, D. S., Darmalaksana, W., & Ramdhani, M. A. (2018). Systematic design of expert system using unified modelling language. In *IOP Conf. Ser. Mater. Sci. Eng* (Vol. 288, No. 1, p. 012047).
- Padmini, K. J., Perera, I., & Bandara, H. D. (2016). Applying agile practices to avoid chaos in User Acceptance Testing: A case study. In *2016 Moratuwa Engineering Research Conference (MERCon)* (pp. 96-101). IEEE.
- Pramitasari, B., & Nurgiyatna, N. (2019). Sistem Informasi Unit Kegiatan Mahasiswa Universitas Muhammadiyah Surakarta Berbasis Web. *Emitor: Jurnal Teknik Elektro*, 19(2), 59-65.
- Rahayu, T. K. (2020, July). Application Report Process Of Islamic School Based On Pesantren Boarding Using Waterfall Model. In *Journal of Physics: Conference Series* (Vol. 1569, No. 2, p. 022025). IOP Publishing.
- Setyawati, R. K. (2018). Efisiensi Pengelolaan Dokumen Berupa Surat Studi Kasus: PT Astra International TBK. *Jurnal Administrasi dan Kesekretarian*, 2(2), 138-149.
- Solanki, N., Shah, D., & Shah, A. (2017). A Survey on different Framework of PHP. *International Journal of Latest Technology in Engineering, Management & Applied Science (IJLTEMAS)*, 6(6), 155-158.
- Vidal-Silva, C., Jiménez, C., Madariaga, E., & Urzúa, L. (2020). Applying PHP Codeigniter For Easy Web Development.
- Yang, L., & Cao, L. (2016). The Effect of MySQL Workbench in Teaching Entity-Relationship Diagram (ERD) to Relational Schema Mapping. *International Journal of Modern Education and Computer Science*, 8(7), 1.